Optimization of Network Protocol Options by Reinforcement Learning and Propagation

FIELD OF THE INVENTION

[0001] The embodiments of the invention relate generally to the field of network communication and, more specifically, relate to optimization of network protocol options by reinforcement learning and propagation.

BACKGROUND

[0002] Trivial file transfer protocol (TFTP) is a simple user datagram protocol (UDP)-based file transfer program that is frequently used in pre-boot environments. For example, TFTP is widely used in image provisioning to allow diskless hosts to boot over the network.

[0003] TFTP provides extensive options, such as block size of data packets and multicast provisioning, which may be applied in order to achieve better performance. For instance, a larger value block size may result in better transfer performance (e.g., a session with the block size of 32KB results in a 700% increased performance gain over a session with the block size of 512B in certain 100Mbps environments). Multicasting enables simultaneous provisioning to multiple clients.

[0004] When a TFTP server receives requests from clients, simple negotiations are conducted in which the TFTP server may select appropriate option values as responses. After the negotiation, TFTP sessions are created and the files are transferred according to the selected options of the sessions. However, TFTP option selection presents problems in the area of optimizing and propagation of these options in different network environments for performance enhancement. The effectiveness of the TFTP options is highly dependent on the specific network environments. Some affecting factors on performance include, but are not limited to: network topology, switches and their configurations, network drivers, and implementation of the TFTP clients.

[0005] In some cases, TFTP options that could lead to high performance in some environments may be risky in other environments, possibly even causing failures. One example is that a single session of a block size of 32KB may fail on one type of switch, while a block size of 16KB may succeed on the same switch with acceptable performance. Another example is that a single multicast session of a block size of 32KB on an older driver version of a certain Ethernet adapter in a 1Gbps environment may fail, while reducing the block size or replacing an updated version of the driver will succeed. These issues become more serious when the environments are complicated.

[0006] For instance, complicated environments may include infrastructures having connectors with hubs, a mix of both 1Gbps connections and 100Mbps connections, implementations of UDP multicast of different switches, multiple sessions occurring simultaneously but starting and ending at different times, specific TFTP clients not perfectly implemented due to pre-boot limitations, etc. There are no obvious rules or guidelines that uniformly work in these different environments. Therefore, under current TFTP implementations, it is difficult for a TFTP server to make optimal decisions during option negotiation that can both achieve a high performance and ensure success of a file transfer.

BRIEF DESCRIPTION OF THE DRAWINGS

[0007] The invention will be understood more fully from the detailed description given below and from the accompanying drawings of various embodiments of the invention. The drawings, however, should not be taken to limit the invention to the specific embodiments, but are for explanation and understanding only.

[0008] **Figure 1** is a block diagram of one embodiment of an exemplary network system to perform embodiments of the invention;

[0009] **Figure 2** is a block diagram of one embodiment of a network environment for providing optimal option selection for trivial file transfer protocol (TFTP);

[0010] **Figure 3** is a block diagram of one embodiment of an application of option optimization using reinforcement learning;

[0011] **Figure 4** is a flow diagram depicting a method of one embodiment of the invention; and

[0012] **Figure 5** illustrates a block diagram of one embodiment of an electronic system to perform various embodiments of the invention.

DETAILED DESCRIPTION

[0013] An apparatus and method for optimization of network protocol options by reinforcement learning and propagation are disclosed. Reference in the specification to "one embodiment" or "an embodiment" means that a particular feature, structure, or characteristic described in connection with the embodiment is included in at least one embodiment of the invention. The appearances of the phrase "in one embodiment" in various places in the specification are not necessarily all referring to the same embodiment.

[0014] In the following description, numerous details are set forth. It will be apparent, however, to one skilled in the art, that the embodiments of the invention may be practiced without these specific details. In other instances, well-known structures and devices are shown in block diagram form, rather than in detail, in order to avoid obscuring the invention.

[0015] Embodiments of the present invention describe a method and respective circuit for optimization of network protocol options by reinforcement learning and propagation. More specifically, embodiments of the invention provide a novel approach to trivial file transfer protocol (TFTP) option negotiation and selection using reinforcement learning and propagation.

[0016] **Figure 1** is a block diagram illustrating one embodiment of an exemplary network system to perform embodiments of the invention. System 100 includes a TFTP server 110, a network 120, and a client 130. TFTP server 110 may listen over network 120 for connection requests from client 130. Client 130 may make a connection to the TFTP server 110. Once connected, client 130 and TFTP sever 1100 may communicate via the TFTP. For instance, client 130 may do a number of file manipulation operations such as uploading files to the TFTP server 110, download files to the TFTP server 110, and so on. In other embodiments, one skilled in the art will appreciate that a server other than a TFTP server communicating via the TFTP (e.g., FTP server) may be utilized.

[0017] Additionally, TFTP server 110 and client 130 may further enter into option negotiations. During option negotiations, options to enhance and modify the functionality of the TFTP may be selected and enacted between the TFTP server 110 and client 130. Embodiments of the invention provide a novel approach for the optimum selection of protocol options during option negotiation by using reinforcement learning and propagation.

[0018] **Figure 2** is a block diagram illustrating one embodiment of a system 200 for providing optimal option selection for TFTP. In one embodiment, a TFTP server 210 interacts with an environment 230 using a trial-and-error strategy by providing different options. In one embodiment, the environment 230 includes a file transfer component 240 of the TFTP server 210, along with a network environment 235 (switches, network drivers, etc.) and one or more TFTP clients 220. The option negotiation component 215 of TFTP server 210 is outside of and interacts with the environment 230.

[0019] In one embodiment, the TFTP server 210 receives performance feedback for the different options as rewards, and improves its decision-making policy for option negotiation based on these past experiences and resulting rewards. In some embodiments, the TFTP server 210 may optionally upload the decision-making policy along with the observed configurations of the specific environment to a centralized place (e.g., an electronic library). Other TFTP servers 210 may then download the resources and use the policy for the most similar environment to start their own trial-and-error learning process. In some embodiments, option negotiation via a decision-making process in uncertain environments is accomplished by applying a Q-learning method.

[0020] In one embodiment, an option negotiation component 215 of the TFTP server 210 may be utilized as an intelligent agent that interacts with the environment 230. The option negotiation component 215 provides the trial options for various environments 230 and receives the rewards as feedback. The option negotiation component 215 then utilized reinforcement learning to come to the optimal option selection for any particular environment 230.

[0021] In some embodiments, the option negotiation component 215 may be in a certain state s_t at a time t. The state is used to describe the specific status of the current system, namely the pending file transfer requests and existing transfer sessions along with the options of the sessions. State transitions may occur whenever a new request is received, new sessions are created, or old sessions are ended.

[0022] At state s_t, the option negotiation component 215 may choose an action a_t from the action set allowed in the state $D(s_t)$. For most of the states where there are no pending file transfer requests, only a null action is allowed. For the states where there are new file transfer requests, the action set includes all of the legal options the TFTP server 210 may respond with. At each time step t, a reward r_t is received describing the utility that the option negotiation component 215 obtains. In some embodiments, a reward may refer to the data transferred at that time plus any penalties incurred, such as those caused by a timeout, session failure, etc.

[0023] In one embodiment, the state transitions are assumed to depend on the action probabilistically according to an unknown distribution $P(s_{t+1} | s_t, a_t)$ of the specific network environment. The rewards are assumed to depend on the state the agent resides and the action it takes probabilistically according to an unknown distribution $P(r_{t+1} | s_t, a_t, s_{t+1})$of the specific network environment.

[0024] The goal of the option negotiation component 215 is to decide appropriate actions to maximize the performance of a file transfer, i.e., to choose appropriate actions to maximize the discounted returns during an infinite long run. This may be demonstrated as:

$$r^{(t)} = \lim_{T \to \infty} \sum_{\tau=0}^{T} \gamma^{\tau} r_{t+\tau} \cdot$$

[0025] In one embodiment, in order to resolve the problem, a Q-function may be introduced that is the expected return of an action *a* at a state *s* with respect to a policy π as:

$$Q^{\pi}(s,a) = E_{\pi}(R^{(t)} \mid S_t = s) = E_{\pi}(\sum_{\tau=t+1}^{\infty} \gamma^{\tau-t-1} R_{\tau} \mid S_t = s, A_t = a),$$

The policy π denotes the probability distribution of choosing actions at the various states. Capital letters, such as *S*, *A*, are used to denote the random variables, and lower case letters, such as *s*, *a*, are used to denote the value of the random variables.

[0026] The Q-function of the optimal policy π^* satisfies the following Bellman optimal equation:

$$Q^{*}(s,a) = \sum_{s'} P_{ss'}^{a} [R_{ss'}^{a} + \gamma \max_{a'} Q^{*}(s',a')],$$

where,

$$P_{ss'}^{a} = P(S_{t+1} = s' \mid S_t = s, A_t = a),$$

and,

$$R_{ss'}^{a} = E(R_{t+1} \mid S_t = s, A_t = a, S_{t+1} = s') = \sum_{r_{t+1}} r_{t+1} P(R_{t+1} = r_{t+1} \mid S_t = s, A_t = a, S_{t+1} = s').$$

[0027] The Q-learning algorithm is a standard approach of reinforcement learning that iteratively calculates the value functions of the optimal policy. Under the Q-learning algorithm, let $\hat{Q}^*(s,a)$ denote the estimated Q function of the optimal policy. These values may then either be stored as a lookup table, or approximated by functions $h(s,a,\mathrm{w})$ with w as parameters (e.g., a linear function of features implied in the states *s* and the actions *a*, or more sophisticated function approximators).

[0028] In one embodiment, the Q-learning algorithm works as follows:

1. Initialize $\hat{Q}^*(s,a)$.

2. $t \leftarrow 0$, $k \leftarrow 1$, start from s_0.

3. Select an action a_t according the distribution

$$P(A_t = a_t \mid S_t = s_t) \propto k^{\hat{Q}^*(s_t,a_t)},$$

and transit to the state s_{t+1}, and receive the immediate reward r_{t+1}.

4. Update the estimated Q function with a sample backup strategy for the Bellman optimal equation

$$\hat{Q}^*(s_t,a_t) \leftarrow \hat{Q}^*(s_t,a_t) + \alpha[r_{t+1} + \gamma \max_{a_{t+1}} \hat{Q}^*(s_{t+1},a_{t+1}) - \hat{Q}^*(s_t,a_t)].$$

5. Increase k and $t \leftarrow t+1$.

6. If the terminate condition is not met, go back to step 2.

7. Optionally retrieve the configurations of the environment and upload the policy (estimated Q function) to a centralized environment.

[0029] **Figure 3** is a block diagram of one embodiment of the application of option optimization using reinforcement learning, such as the Q-learning algorithm, in a system 300. The components of system 300 interact together to utilize various embodiments of the invention. The components of system 300 include an option provider 310, a file transfer component 320, and a Q-function update component. In one embodiment, these components are included as part of TFTP server 210, described with respect to **Figure 2**.

[0030] In one embodiment, option provider 310 receives file transfer requests. Option provider may associate the environment of the file transfer requests with, for example, Q values related to a Q-learning algorithm. Option provider 310 may then select options for the environment based on the Q values. These selected options, as well as the file transfer requests, are sent to the file transfer component 320.

[0031] File transfer component 320, in turn, transfers data associated with the file transfer requests. File transfer component 320 also sends feedback, or rewards, to Q function update component 330. Q function update component may modify its Q values that is provides to option provider 310 based on the rewards received from file transfer component 320.

[0032] In some embodiments, the components of system 300 utilize a Q-learning algorithm, such as that described above. In the initialization stage (e.g., step 1) of the above algorithm, the initial Q function values may be randomized if there is no further information available. However, if the server is able to download resources from the centralized environment, the server may select the policy of the most similar environment by comparing the observed configurations to initialize the Q function.

[0033] When the values of the estimated Q function are stored with a lookup table, the estimated Q function converges to the values of the optimal policy when the parameters are controlled in an appropriate manner. The action selected in step 2 of the algorithm, may be optimal when *k* gets larger after a certain number of iterations.

[0034] **Figure 4** is a flow diagram illustrating a method of one embodiment of the invention. Process 400 provides a method for optimization of network protocol options with reinforcement learning and propagation. The process 400 begins at processing block 410 where a learning component of a TFTP server interacts with clients, as well as with the environment, by conducting different trials of various TFTP options in different states. Then, at processing block 420, the learning component of the TFTP server receives performance feedback for these trials as rewards.

[0035] At processing block 430, the learning component of the TFTP server utilizes the past trials and resulting rewards to improve its decision-making policy for option negotiation. In some embodiments, a reinforcement learning algorithm is used to improve the decision-making policy. In one embodiment, the reinforcement algorithm may be a Q-learning algorithm.

[0036] At processing block 440, the learned policies for various option implementation decisions are uploaded, along with the observed configurations of the environment, to a centralized place (e.g., an electronic library). Then, at processing block 450, other TFTP servers may then download the resources and use the policy of the most similar environment as the initial point to start a new learning process in their environments.

[0037] One skilled in the art will appreciate the embodiments of the present invention may be applied to communication protocols other than TFTP, and the present descriptions are not intended to limit the application of the various embodiments to solely TFTP.

[0038] In some embodiments, components of the TFTP server or other clients may utilize various electronic systems to perform embodiments of the invention. The electronic system 500 illustrated in **Figure 5** is intended to represent a range of electronic systems, for example, computer systems, network access devices, etc. Alternative systems, whether electronic or non-electronic, can include more, fewer and/or different components.

[0039] Electronic system 500 includes bus 501 or other communication device to communicate information, and processor 502 coupled to bus 501 to process information. In one embodiment, one or more lines of bus 501 are optical fibers that carry optical signals between components of electronic system 500. One or more of the components of electronic system 500 having optical transmission and/or optical reception functionality can include an optical modulator and bias circuit as described in embodiments of the invention.

[0040] While electronic system 500 is illustrated with a single processor, electronic system 500 can include multiple processors and/or co-processors. Electronic system 500 further includes random access memory (RAM) or other dynamic storage device 504 (referred to as memory), coupled to bus 501 to store information and instructions to be executed by processor 502. Memory 504 also can be used to store temporary variables or other intermediate information during execution of instructions by processor 502.

[0041] Electronic system 500 also includes read only memory (ROM) and/or other static storage device 506 coupled to bus 501 to store static information and instructions for processor 502. Data storage device 507 is coupled to bus 501 to store information and instructions. Data storage device 507 such as a magnetic disk or optical disc and corresponding drive can be coupled to electronic system 500.

[0042] Electronic system 500 can also be coupled via bus 501 to display device 521, such as a cathode ray tube (CRT) or liquid crystal display (LCD), to display information to a computer user. Alphanumeric input device 522, including alphanumeric and other keys, is typically coupled to bus 501 to communicate information and command selections to processor 502. Another type of user input device is cursor control 523, such as a mouse, a trackball, or cursor direction keys to communicate direction information and command selections to processor 502 and to control cursor movement on display 521. Electronic system 500 further includes network interface 530 to provide access to a network, such as a local area network.

[0043] Instructions are provided to memory from a storage device, such as magnetic disk, a read-only memory (ROM) integrated circuit, CD-ROM, DVD, via a remote connection (e.g., over a network via network interface 530) that is either wired or wireless providing access to one or more electronically-accessible media, etc. In alternative embodiments, hard-wired circuitry can be used in place of or in combination with software instructions. Thus, execution of sequences of instructions is not limited to any specific combination of hardware circuitry and software instructions.

[0044] Embodiments of the invention provide numerous advantages over prior art solutions, including: (1) dynamically deciding TFTP option to optimize the network performance according to the environment; (2) adaptive, self-learning approach for option optimization; and (3) information propagation of learned strategies in different environments for future reuse.

[0045] In addition, embodiments of the invention provide a self-learning, self-adapting, and self-distributing system seamlessly integrated into standard TFTP without impacting current protocol options and capabilities. One skilled in the art will appreciate that embodiments of the invention may potentially be applied to other network transportation protocols, such as file transfer protocol (FTP).

[0046] Whereas many alterations and modifications of the present invention will no doubt become apparent to a person of ordinary skill in the art after having read the foregoing description, it is to be understood that any particular embodiment shown and described by way of illustration is in no way intended to be considered limiting. Therefore, references to details of various embodiments are not intended to limit the scope of the claims, which in themselves recite only those features regarded as the invention.